

REMARKS

1. Rejections Under 35 U.S.C. § 103(a): Claims 1-8 and 11-20 have been rejected under 35 U.S.C. §103(a) as being unpatentable over US 6,167,427 (“‘427”) in view of US Application 2002/0078213 (“‘8213”). In asserting ‘427, it was recognized that ‘427 does not explicitly teach about the application being implemented in an object-oriented JAVA base environment.

1.1 The Claims

In response to the combination of references on which the rejections are based, it is noted that the claimed method for load balancing in a JAVA based environment (claim 1) defines executing an application including a first service module and a control module. The control module includes application-specific strategies for the application. The claimed load-balancing includes generating of second service modules is based on the application’s application-specific policies. It is also noted that the claimed method of claim 7 includes a control module that includes application-specific load-balancing policies for the application, in a JAVA code form. It is also noted that the claimed method for moving an application of claim 15 defines executing a first service module and a control module of the application on a first server, the control module having application-specific policies for the application, and sending a message from the control module to an executive runtime module, the message requesting the executive runtime module to move the first service module to a second server according to the application-specific policies.

1.2 Lack Of Motivation To Combine 8213 with ‘427

The following remarks indicate that the teachings of 8213 teach away from the combination, such that one skilled in the art would not be motivated to combine 8213 with ‘427. First, the cited paragraph 0052 emphasizes a system management framework to

manage the resources in the managed region. The Java runtime environment is generally referenced, and neither the references in the cited paragraph 0052 nor the cited paragraph 0019 to platform independence describes platforms in relation to the claimed application-specific policies.

In support of 8213 teaching away from the combination, reference is made to 8213 at paragraph 0075. Initially, it is clearly stated that the application “requires some type of insulation from the specifics of the operations of the gateways” that implement requests from the application for use of resources. As direct evidence that there is no claimed application-specific policy in the 8213 application, paragraph 0075 states that the NELS (and thus not an application-specific policy) “balance[s] the load between the gateways to handle resources...”. This is a teaching away from the claimed application-specific policies.

Consistent with this evidence of 8213 teaching away from the claimed application-specific policies, reference is also made to paragraph 0096. There, it is stated that “a system or network administrator may predetermine certain criteria to be applied to the resources that impinge on the availability of the particular resources for certain users during certain schedules.” In impinging on the availability of resources for a certain user (i.e., application), it is the system or administrator, and not the application (user), that defines the policy for resource availability. Moreover, in paragraph 0109, the gateway, and not a claimed application-specific policy, “determines whether or not the lease should be terminated...” (step 794), and if so, “...then the application is notified...”. Alternatively, when the lease is not terminated, paragraph 0126 makes it clear that the 8213 application does not control altering “the manner in which active action objects” are successfully completed.

The reasoning of the rejection (page 3, last paragraph) in support of 8213 easily accommodating future changes in network resource, based on the code being protocol

independent and network-route unaware, does not take into account these clear statements describing non-application-specific policies governing use of system resources.

1.3 Review of 8213 As A Whole

In further response, it is respectfully submitted that 8213 should be reviewed as a whole, and not merely to the cited paragraphs 0019 and 0052. These above-noted teachings other than paragraphs 0019 and 0052 would make it clear to one skilled in the art that in the context of that Java runtime environment, a premise of 8213 is to not allow the applications to have the claimed application-specific policies, thus to not have the claimed application-specific policies in the application govern load balancing. This is clear from all three of paragraphs 0096 (e.g., administrator may predetermine certain criteria for the application), 0109 (gateway determines whether or not the lease should be terminated); and 0126 (application does not control altering the manner in which active action objects are successfully completed). In view of these direct 8213 teachings, allowing the applications to have the claimed application-specific policies would require a fundamental change in the way the determination is made by 8213 as to whether and how resources will be made available to the application, i.e., for the claimed load-balancing based on the claimed application-specific policies.

It would not be proper to pick and chose only the paragraph 0052 teaching without also giving due consideration to the 8213 teaching of not using an application-specific policy for resource availability, for example. To establish a *prima facie* case of obviousness, there must be some suggestion or motivation, either in the reference or in the knowledge generally available, to one having ordinary skill in the art, to combine the references. That suggestion or motivation is not present when one reference teaches away from the combination. When the required due consideration is given, it is respectfully submitted that one would be taught

by 8213 to have the system or administrator or gateway govern the policies, and would be lead away from the claimed invention in which the application-specific policies govern the provision of application-specific load-balancing strategies to the JAVA platform. Thus, it is respectfully submitted that it is improper to combine 8213 with '427. Accordingly, withdrawal of the 8213 reference is requested.

1.4 No Prima Facie Case Is Properly Applied To The Amended Claims Based On the
Combined References

1.4.1 Admitted Deficiency Of '427

The references when combined must teach or suggest all the claim features. Without the 8213 reference, in view of the admitted deficiency of the '427 reference (page 3, 2 lines above last paragraph), the asserted '427 alone does not establish a prima facie case of obviousness as admitted by the rejections. Accordingly, withdrawal of the rejection of the pending claims is respectfully requested.

1.4.2 The '427 Replication Service Does Not Teach The Claimed Features Of Load
Balancing Based On Application-Specific Policy In Application Control Module

As a further basis for not applying the rejections to the present claims, the following remarks show that even if combined (contrary to the above showing), the combined references would not teach the claimed inventions. In detail, the rejection asserts that in respect to the "executing an application ..." clause of claim 1, the "Replication Service" of '427 is a claimed control module, **"wherein the control module includes application-specific policies"** (policy specific to loading) for the application. C4, L55-67, C6, L15-20, and C6, L40-50 are referenced.

For reference, it is noted that claim 1 defines executing a JAVA application having a first service module and a control module, wherein the control module includes application-specific load-balancing policies for the application.

For further reference, it is noted that claim 7 defines an application having application-specific strategies for use in a JAVA environment, wherein the application comprises a control module in communication with the plurality of service modules, wherein the control module includes application-specific load-balancing policies for the application.

For further reference, it is noted that claim 15 has been clarified and now more clearly defines the method for moving an application. The operation of executing the application provides that the service module and control module are of the application. Further, claim 15 defines the sending of a message from the control module to an executive runtime module, the message requesting the executive runtime module to move the first service module to a second server according to the application-specific policies.

The use of the application-specific policies for this purpose is supported by the description at page 29, lines 10-12 (see “control module can make a decision”), and page 29, lines 19-22, for example.

1.4.2.1 Review of ‘427 As A Whole

In regard to the rejection asserting the “Replication Service” of ‘427 as a claimed control module, wherein the control module includes application-specific policies, e.g., a policy specific to loading, for the application, it is respectfully submitted that it is proper to review ‘427 as a whole, including review of the complete description of the replication service.

The rejection cited C4, L55-67. Here, the replicating by '427 is described in terms of a server that runs an object (server p) and the server (server q) that will replicate the object. Importantly, the system is said to be able to balance the load in two ways. Neither of those ways is a claimed way in which the application itself to be replicated directs the replication according to the application's own application-specific load balancing policy.

This observation is clear from the following other parts of '427. For example, reference is made to the C5, L55-60 description of objects being created and placed on a hosting server. Importantly, the hosting servers are separate from the replication service. First, at the cited C6, L16 it is stated that the "hosting servers periodically report their load to the replication service unit 130." If the hosting servers made their own decisions as to load-balancing, there would be no need to report their loads to the replication service unit 130. Further, because the hosting servers are separate from the replication service, the structure of the replication servers is not part of the applications hosted on the hosting servers. Thus, any application hosted by the hosting servers does not include any load-balancing control function of the separate replication service.

As to the hosting servers periodically reporting their load to the replication service, the cited C6, L17-18 states that "The replication service uses these reports [from the hosting servers] to migrate or replicate objects...". This does not state that the replication service uses application-specific policies of or from the hosting server to do such migration or replication. Rather, it is current load data (and not application-specific policies) that is sent from the hosting servers to the replication service.

As a further indication that any application hosted by the hosting servers does not include any control function of the replication service, and that any control function of the

replication service does not include any control function of application-specific policies of the application, the cited C6, L40-50 is prefaced by a description (C6, L36+) of a scalable distributed architecture for the replication service 130. Reference is made to “a policy for replica placement” being based on the objective of load distribution. Starting at C6, L49 a structure for the replication service 130 is described. Such service is a hierarchy of replicator servers (C6, L51). At L57, this hierarchy is said to perform analyses of load information, and (L57-58) generates control messages in re creating and deleting object replicas. Thus, contrary to the assertion in the rejection, the ‘control’ by the hierarchy of replicator servers is part of the separate replication service, and is not the claimed control module that is part of the application and that has the application-specific policies.

In view of the above-described hosting servers, which host (and thus run) the application objects, and in view of these replication servers that are separate from the hosting servers, in ‘427 there is no application hosted by one of the hosting servers that runs on such replication servers. As a result, there is no application control module that runs on the replication servers. Moreover, because this hierarchy of replication servers itself performs the analyses of load information received from the hosting servers without receiving hosting server policies specific to the applications hosted by the hosting servers, and because this hierarchy of replication servers itself generates (C6, L57-58) control messages in re creating and deleting object replicas, the description at the cited C6, L15-50 shows that (1) the ‘427 replication service does not use application-specific policies of the hosted application on the hosting servers to perform the replication that is performed by the replication service, and (2) the replication servers are not the claimed control module that is part of an application that runs on one of the hosting servers.

Considering '427 further, in the C7 embodiment (L59+) one replicator does load distribution in its subtrees (L65). There are three parameters on which the protocol for replication is based (C8, L1-5). It is respectfully submitted that none of these parameters is application-specific as claimed. Rather, the parameters include:

- (1) system-wide (C8, L3), which is not application-specific;
- (2) those related to all of the server hosts "p"; see C8, L11-14, where it is made clear that the formula at C8, L8 is not based on each application at each hosting server; instead the formula is based on the deletion threshold $u_{sub\ p}$ being "adjusted for every [replication] server such that if object x qualifies for deletion on one [replication] server under a certain request rate, it would qualify for deletion on all [replication] servers under about the same rate". Importantly, it would be a system-wide policy (and not a claimed application-specific policy) that would make such adjustment of the deletion threshold $u_{sub\ p}$ so as to qualify the one object application for deletion on all replication servers under about the same rate; and
- (3) another system-wide parameter, here a system-wide stability factor "s" (C8, L15), that is clearly not application-specific.

Moreover, at C8, L41+ '427 further indicates that in response to all of the load reports, it is the replicator R that executes the protocol. The description of this protocol indicates that it is a system-wide protocol, and not an application-specific protocol of a control module of the application. It is the continued description of this system-wide protocol that is the subject of the cited C9, L 15-20. Thus, it is the system-wide protocol of C8, L41+ that effects the cited C9, L5+ load-redistribution between some hosting servers (C9, L6).

In review, there is nothing in the above-reviewed '427 disclosure to support the assertion in the rejection that the "Replication Service" of '427 is a claimed control module,

"wherein the control module includes application-specific policies" (policy specific to loading) for the application. As a result, it is left to the cited C9, L 15-30 (cited against claim 4) to be the basis for the rejection. However and significantly, as to the C9, L10-12 description of use of different criteria for additional redistribution, it is not application-specific criteria that is described. Rather, the different criteria is selected by R, which is not a hosting server that hosts the application that has the application-specific policies. Rather, R is one of the replication servers, and it is R that advises its hosting replication server (C9, L13-16) as to which hosting server to use if R's parent replication server decides to redistribute the load. Moreover, this description does not indicate any criteria other than the above-distinguished system-wide protocol on which R's parent replication server would make that decision.

It is respectfully submitted that this discussion of '427 clearly indicates that (1) the applications on the hosting servers are separate from the replication service, (2) the replication servers do not act in response to application-specific policies of the application objects hosted on the hosting servers, (3) the structure of the replication servers is not part of the applications hosted on the hosting servers, and (4) no control function of the replication service is part of any application hosted by the hosting servers.

Thus, in terms of claim 1, '427 does not describe:

executing a JAVA application having a first service module and a control module, wherein the control module includes application-specific load-balancing policies for the application...

generating a second service module, using the first service module, the generating of the second service module being based on the application-specific policies of the application, in response to the sensed utilization of system resources;

Thus, in terms of claim 4, '427 does not describe a method as recited in claim 1,
wherein:

the application-specific policies include a specific server on which to generate the second service module.

Thus, in terms of claim 6, '427 does not describe

a method wherein the specific server is selected based on the application-specific policies of the control module.

Thus, in terms of claim 7, '427 does not describe:

An application having application-specific strategies for use in a JAVA environment, comprising:....a control module in communication with the plurality of service modules, wherein the control module includes application-specific load-balancing policies for the application,

Thus, in terms of claim 15, '427 does not describe a method for moving an
application within a JAVA environment, comprising the operations of:

executing a first service module and a control module of the application on a first server, the control module having application-specific policies for the application,

sending a message from the control module to an executive runtime module, the message requesting the executive runtime module to move the first service module to a second server according to the application-specific policies;

Respectfully, in the context of the present independent claims 1, 7, and 15, and of the claims dependent thereon, it is submitted that it would not be proper to continue to assert that the "Replication Service" of '427 is a claimed control module (all claims), or to continue to assert that the '427 control of the replication service is part of a claimed application, or to continue to assert that the '427 control of the replication service includes application-specific policies for the claimed application (all claims). As a result, since '427 was cited as teaching these features, and the foregoing detailed review of '427 shows these claimed features to be

missing from '427, the asserted '427 taken with 8213 do not establish a prima facie case of obviousness.

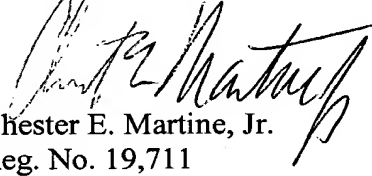
Therefore, Applicants respectfully request the Examiner to allow the pending claims.

Conclusion

In view of the foregoing, Applicants respectfully submit that all the pending claims 1-8, and 10-20 are in condition for allowance. Accordingly, a Notice of Allowance is respectfully requested.

Respectfully submitted,

MARTINE PENILLA & GENCARELLA, L.L.P.



Chester E. Martine, Jr.
Reg. No. 19,711

Martine Penilla & Gencarella, LLP
710 Lakeway Drive, Suite 200
Sunnyvale, California 94085
Telephone: (408) 774-6908
Customer Number 32291